

Перспективи застосування програмного забезпечення для генерування акордів політерцієвої структури та їх модифікацій

Андрій Бондаренко

Київський національний університет культури і мистецтв, Київ, Україна

Мета статті — проаналізувати можливості програмного забезпечення для генерування та вивчення різновидів і модифікацій політерцієвих акордів. *Результати дослідження*. Виявлено, що вчення про акорди політерцієвої структури не знайшло належного висвітлення в науковій і навчально-методичній літературі. З'ясовано, що програмне середовище дає змогу розробити програмне забезпечення, здатне генерувати акорди політерцієвої структури та їх модифікації (обернення та різні розташування). Показано, що кількість можливих політерцієвих акордів залежить від кількості звуків і зростає від 4 (для тризвуків) до 37 (для 10-звучних акордів) та знову спадає до 4 (для 12-звучних акордів). Показано, що нонакорд може приймати 120 модифікацій, ундецимакорд — 720, терцдецимакорд — 5040. Проте як для зручності виконавця, так і зі стилістичних міркувань у процесі розробки програмного забезпечення доцільно передбачати можливість обмеження діапазону та інтервальної структури заданого акорду. *Наукова новизна*. Поглиблено уявлення про обернення та розташування політерцієвих акордів, уперше застосовано програмне середовище C++ для розробки програмного забезпечення, що генерує акорди складної структури, вперше представлено повну таблицю модифікацій нонакордів і таблицю усіх можливих 12-звучних політерцієвих акордів. *Висновки*. Показано, що розробка відповідного програмного забезпечення може поглибити знання митців і музикознавців щодо можливостей акордів політерцієвої структури, полегшити їх роботу, відкрити нові горизонти творчості. Отримані результати доцільно використати для оновлення навчально-методичної літератури з гармонії, яка нині недостатньо розкриває теорію політерцієвих. Водночас зроблено висновок про те, що програмне забезпечення не може замінити сам акт творчості, який потребує від митця емоційного включення, а також багаторівневого контекстного мислення під час вибору гармонічних засобів.

Ключові слова: гармонія; політерцієві акорди; композиція; штучний інтелект

Для цитування

Бондаренко, А. (2023). Перспективи застосування програмного забезпечення для генерування акордів політерцієвої структури та їх модифікацій. *Вісник КНУКіМ. Серія: Мистецтвознавство*, 48, 57–64. <https://doi.org/10.31866/2410-1176.48.2023.282443>

Вступ

Музичне мистецтво ХХ–ХХІ ст. характеризується прискореним розвитком засобів музичної виразності, що зумовлює її ритмічне, тембральне та гармонічне розмаїття. Поява нових музичних інструментів, розвиток звукозапису та інформаційних технологій впродовж цього періоду відкрили для музикантів нові можливості пошуку тембральних, фактурних, гармонічних рішень; і навпаки — постійно оновлювані запити композиторів спонукають інженерів і програмістів

вдосконалювати технології з метою надати ще більше простору для творчої фантазії митців. Акорди політерцієвої структури знайшли широке застосування у джазовій музиці в середині ХХ ст., проте їх теоретичне осмислення досі не потрапляє до відповідної фахової літератури, ймовірно, через великий обсяг аналітичного матеріалу та складність його обробки. Ця обставина спонукає нас звернутись до ресурсів інформаційних технологій, що можуть надати додаткові потужності для вивчення їх структурних особливостей і застосування в музичній практиці.

Аналіз попередніх досліджень. Застосування комп'ютерних технологій у музичному мистецтві та музикознавстві останніми роками набирає популярності, особливо в закордонних дослідженнях. Обмежимося аналізом досліджень комп'ютерного генерування музичного матеріалу.

В Україні перші дослідження застосування комп'ютера для створення музичних структур належать К. Фадєєвій (2007) та І. Ракуновій (2008). У цих роботах йдеться про алгоритмічну музику, у створенні якої комп'ютер відіграє допоміжну роль, а саме — генерує певні послідовності нот, ритмічна і звуковисотна сторона яких зумовлена обраними композиторами алгоритмами. Зокрема, І. Ракунова (2008) аналізує твір А. Загайкевич «Повітряна механіка» для флейти, кларнета, перкусії, фортепіано, скрипки та віолончелі, нотний текст якої значною мірою створено за допомогою комп'ютерного програмного забезпечення.

Останнім часом набуває все більшої популярності поняття штучного інтелекту — програмного забезпечення, що покликане «імітувати поведінку людини» (Бхатті, 2022). Серед останніх робіт українських авторів, які досліджують штучний інтелект (ШІ) у музичному мистецтві, назвемо роботи О. Осадчої (2020) та А. Чібалашвілі (2021), в яких згадуються такі програми для створення музики, як Alice, Flow Machines Composer, Endel, Choral, Emily Howell, а також О. Кравчука (2023), який досліджує результати звернення до ШІ у творчості українських рок-гуртів «Океан Ельзи» та The Strechers з погляду показників популярності й доходить висновку щодо певних вад штучного інтелекту, зокрема таких, як «відсутність емоційної та емоційно виразної складової» або «неповне розуміння музичного контексту» (с. 84).

Згадаємо деякі роботи закордонних авторів. Г. Зурич (Zulić, 2019) досліджує застосування комп'ютера AIVA, що генерує фортепіанну музику, сприймаючи як вхідну інформацію рухи тіла актора (хореографа). Р. Данненберг (Dannenberg, 2020) розглядає проблеми машинного навчання для розв'язання задач комп'ютерного акомпанементу, композиції та синтезу звуку.

Менш дослідженим лишається питання комп'ютерного генерування окремих гармонічних структур, що могли бути використані в музичних творах. Наше рішення зосередитись на дослідженнях політерцієвих акордів зумовлено й тим, що сама теорія побудови та модифікацій таких акордів залишається недостатньо розробленою. Зокрема, на необхідності оновлення навчально-методичної літератури з гармонії з метою охоплення вивчення сучасних гармонічних засобів наголошувала Є. Морєва (2018).

Мета статті — проаналізувати можливості програмного забезпечення для генерування та вивчення різновидів і модифікацій політерцієвих акордів.

Методи дослідження — моделювання звуковисотних структур у програмному середовищі C++ з подальшим нотографічним представленням результату та його оцінкою.

Результати дослідження

Як відомо, в класичній гармонії акорди будуються по терціях, і тому в загальній гармонічній теорії називаються акордами терцієвої структури. Зазвичай класична гармонія оперує акордами з трьох або чотирьох звуків (тризвуки та септакорди), натомість акорди терцієвої структури з 5 і більшою кількістю звуків почали активно використовуватися лише у ХХ ст. Ці акорди заведено називати політерцієвими (з грец. *πολύ* — багато), в англомовній термінології є термін *extended chord* (дослівно — розширений акорд).

У підручниках гармонії теорія політерцієвих акордів майже не розглядається, що породжує певний інформаційний вакуум як у сфері музично-теоретичного аналізу, так і творення музики. Навіть найпростіший із політерцієвих акордів — нонакорд, висвітлюється у підручниках обмежено і суперечливо. Наприклад, у виданні І. Дубініна (1982) розглядаються лише нонакорди V, II і IV ступенів в основному вигляді і вказується, що «обернення нонакордів вживаються значно рідше» (с. 46). Я. Таргош (Targosz, 2004) розглядає лише домінантовий нонакорд і вважає можливими лише I і 3-є обернення, а також додатково висуває вимогу про те, що «нона має бути вище за основний тон на відстані щонайменше нони» (с. 173). М. Лемішко (2010) вважає, що більшість нонакордів «не вписується в [гармонічну] систему» (с. 112), і, за винятком нонакордів V і II ступенів займають «маргінальне місце», а обернення нонакордів — «не набули самостійного значення» (с. 113). Фактично, позиція усіх згаданих авторів майже не відрізняється від положень, викладених у редакціях видання М. Римського-Корсакова (1948) ще середини ХХ ст. (наприклад, українське перевидання).

Більш сміливий підхід знаходимо у трактаті А. Шенберга, вперше опублікованому в 1910 році. Зважаючи на відсутність, на жаль, українського перекладу цієї цінної роботи, дозволимо собі процитувати розгорнуті пояснення композитора щодо теорії нонакордів: «Наскільки я знаю, найважливіше заперечення проти нонакордів полягає в тому, що їх обернення — не практичні. Підозрюйте також дурну перешкоду, що дев'ятий акорд нелегко представити в чотириголосному написанні; зара-

ди дев'ятого акорду нам знадобиться п'ять-шість голосів. Можна, звичайно, знехтувати аналогією з оберненнями септакордів, але теорія має тенденцію кожного разу, коли не має прикладу для чогось, оголосити це поганим або принаймні неможливим. Теорія теж охоче каже: нонакорди не з'являються в оберненнях, отже обернення нонакорду — погані; або: [...] їх просто не існує. Звичайно, було б неправильно, якби теоретики винайшли обернення нонакордів, а не чекали, поки це зроблять композитори. Теорія не може бути лідером; вона має стверджувати, описувати, порівнювати та організовувати. Тому обмежусь тим, що даю композиторам і майбутнім теоретикам декілька стимулів подальшого розширення системи та утримаюсь від систематизації...» ((Schoenberg, 1910, p. 345); *переклад з англ. — А.Б.*).

На наш погляд, з метою «практичного стимулу» «композиторам і майбутнім теоретикам» на сучасному етапі розвитку музичного мистецтва було б доцільно вирішити принаймні дві задачі:

– Генерування всіх можливих політерцієвих акордів із довільною кількістю різних звуків.

– Генерування всіх можливих модифікацій заданого акорду.

Розв'язання зазначених задач є доволі простим для акордів із кількістю звуків до 4, натомість для акордів із кількістю звуків 5 і більше потребує значної кількості логічних або математичних операцій, що і спонукає нас застосувати програмне забезпечення. За програмне забезпечення ми обрали програмне середовище C++ як одну з найуніверсальніших мов програмування, тому фрагменти розроблених автором цієї статті кодів нижче подаються саме цією мовою кодування. Для здійснення відповідних обчислень ноти задаються у числових значеннях, що відповідають двом показникам — ступеня (7 ступенів в кожній октаві) та звуковисотності (12 тонів у кожній октаві). Введення даних здійснюється через інтерактивний діалог з користувачем (передбачено конвертацію стандартних буквених позначень звуків у відповідні числові значення). Виведення даних можливе як у вигляді списку нот, так і у вигляді файлу xml, який можна відкрити в нотних редакторах для відображення у вигляді нотного тексту або подальшої обробки.

Сформулюємо першу задачу: знайти усі можливі акорди від заданого звуку із заданою кількістю звуків N, які будуються відповідно до таких умов:

– інтервали між сусідніми звуками становлять малу терцію або велику терцію (3 або 4 півтони);

– жоден звук в акорді не може бути повторений, зокрема енгармонічно рівним звуком (інтер-

вали між звуками акорду не можуть бути складати $12n$ або $12n + 7$, де $n \geq 0$).

Наводимо фрагменти коду, що забезпечують розв'язання цих двох задач:

```
for (int i = 0; i < counter * 2; i += 2)
{
    for (int k = 0; k <= 1; k++)
    {
        A[i + k].pitch[sound + 1] = A[i + k].pitch[sound]
        + k + 3;
        A[i + k].step[sound + 1] = A[i + k].step[sound]
        + 2;
    }
}
bool PitchFilter(int* pitch, int sounds)
{
    for (int i = 0; i < sounds; i++)
        for (int j = 0; j < sounds; j++)
            if (i != j && (pitch[i] - pitch[j]) % 12 == 0)
                return false;
    return true;
}
```

Не вважаючи за можливе в статті навести усі отримані політерцієві акорди, обмежимося найбільш вагомими спостереженнями. Кількість акордів, що може бути побудована для акордів із N звуків, складає від 4 до 37 (табл. 1).

Таблиця 1

Кількість політерцієвих акордів залежно від кількості нот в акорді

Звуків в акорді, N	Інтервал між басом і мелодичним тоном (ступенів)	Кількість можливих акордів
3	5	4 (збільшений, мажорний, мінорний і зменшений тризвуки)
4	7	7 (збільшений мажорний; великий мажорний та мінорний; малий мажорний, мінорний та зменшений)
5	9	12
6	11	21
7	13	36
8	15 (дві октави)	35
9	17 (дві октави + терція)	35
10	19 (дві октави + квінта)	37
11	21 (дві октави + септима)	21
12	23 (три октави + секунда)	4



Рис. 1. 12-звучні політерцієві акорди. Джерело: скриншот автора

Найцікавішим для композиторів зазвичай є останній політерцієвий акорд, що містить усі 12 звуків дванадцятиступеневого звукоряду. Наведемо усі 4 можливі акорди (для зручності прочитання вільно застосовуємо енгармонічні заміни) (Рис. 1):

Розглянемо наступну задачу — генерування усіх можливих модифікацій політерцієвого акорду. Як відомо, будь-який акорд може бути модифікований через зміну порядку розташування звуків. Ми пропонуємо використовувати об'єднувальний термін — модифікації, що означає видозміни акорду через його обернення (місце нижнього звуку займає один з інших звуків акорду) або зміни розташування звуків (змінюється порядок розташування решти звуків).

Загалом кількість модифікацій можна розрахувати за відомою з математики формулою для обчислення перестановок:

$$P_n = n! \quad (1),$$

де P — кількість можливих модифікацій, n — кількість звуків в акорді, а $n!$ — це факторіал числа, що обчислюється добутком усіх чисел, що менші або дорівнюють йому (наприклад, $4! = 1 \times 2 \times 3 \times 4 = 24$).

З цієї формули (1) випливає, що нонакорд має 120 модифікацій, ундецимакорд — 720, а терцдецимакорд — 5040. Звичайно, пошук усіх можливих модифікацій акордів вручну — надзвичайно трудомістка справа, відтак пропонуємо застосувати можливості програмного середовища C++. Для пошуку модифікацій нонакорду необхідно перерахувати усі комбінації із 5 заданих звуків на 5 позиціях, відфільтрувавши комбінації, що містять повторювані звуки. Фрагмент відповідного коду наведено нижче:

```
for (int i = 0; i < 6; i++)
    for (int j = 0; j < 6; j++)
```

```
        for (int k = 0; k < 6; k++)
            for (int l = 0; l < 6; l++)
                for (int m = 0; m < 6; m++)
                    for (int n = 0; n < 6; n++)
                    {
if (i != j && i != k && i != l && i != m && i != n &&
j != k && j != l && j != m && j != n && k != l && k
!= m && k != n && l != m && l != n && m != n)
        {
            Chord[c].key[0] = Initial.key[i];
            Chord[c].key[1] = Initial.key[j];
            Chord[c].key[2] = Initial.key[k];
            Chord[c].key[3] = Initial.key[l];
            Chord[c].key[4] = Initial.key[m];
            Chord[c].key[5] = Initial.key[n];

            c++;
        }
    }
}
```

Отриманий результат буде таким, як показано на Рис. 2.

Зауважимо, що в окремих модифікаціях нона може опинитися як нижче основного тону, так і на відстані секунди від нього, що спонукає говорити про наявність не тільки широких і тісних розташувань акорду, але й звужених (наприклад, d-f-g-a-h). Найвужче розташування охоплюватиме діапазон великої сексти, а найширше — три октави та терцію.

Для ундецимакордів і терцдецимакордів підрахунок здійснюється аналогічно. Проте з практичних міркувань при побудові акордів доцільно вводити додаткові обмеження. Наприклад, обмежившись (зі стилістичних міркувань) лише ундецимакордами, в яких і нона, і ундецима розташовані над основним тоном на відстані більше октави, а також обмежившись акордами, зручни-



Рис. 2. 120 модифікацій великого доміантового нонакорда.

Джерело: скріншот автора



Рис. 3. 18 модифікацій великого мажорного ундецимакорда з ноною та ундецимою, вищими від основного тону, зручними для виконання на фортепіано.

Джерело: скріншот автора

ми для виконання на фортепіано (припускаємо можливість виконати однією рукою акорди діапа-

зоном лише в межах октави), замість 840 можливих акордів, отримуємо лише 18 (Рис. 3).

Висновки

Теорія політерцієвих акордів і їх модифікацій розроблена недостатньо через великий обсяг розрахунків, які проводити вручну обтяжливо. Натомість програмне середовище, зокрема C++, дає змогу розробляти програмне забезпечення, здатне генерувати політерцієві акорди за різними умовами як ладо-гармонічного характеру (різна кількість звуків, інтервальна структура), так і виконавського (обмеження діапазону викликані технічними можливостями виконавця). Подібне програмне забезпечення може, з одного боку, допомогти композиторам і виконавцям у виборі гармонічних засобів, а з іншого — допоможе музикознавцям глибше зрозуміти ладо-гармонічні властивості політерцієвих акордів.

Наукова новизна. Показано, що програмне середовище, зокрема мови C++, дає змогу розробляти програмне забезпечення, здатне генерувати політерцієві акорди за різними умовами, як-от: усі можливі політерцієві акорди із заданою кількістю звуків від заданої ноти; усі можливі модифікації (обернення і розташування) заданого акорду: модифікації заданого акорду з обмеженнями діапазону або інтервальної структури.

Виглядає доцільним *продовжити дослідження* в цьому напрямі, зокрема створити програми, здатні генерувати не тільки окремі акорди, але і їх сполучення. Подальша інтеграція таких програм у нотні редактори може надалі спростити роботу композиторам та аранжувальникам, розкрити нові можливості для творчої фантазії.

Водночас слід зазначити, що розроблені нами програми не покликані замінити чи знецінити творчу роль композитора. Згадані вище вади штучного інтелекту, такі як відсутність емоційного та контекстного мислення, лишаються актуальними, відтак перспективи застосування досліджуваного програмного забезпечення обмежуються лише допомогою митцям і мистецтвознавцям у глибшому розумінні та гнучкішому оперуванні можливостями звуковисотної системи.

Список посилань

- Бабич, І. Р., & Яшина, О. М. (2020). Підходи до розробки штучного інтелекту та їхній вплив на автоматизацію написання музики. *Modern Scientific Researches*, 12(1), 35–39.
- Бхатті, Ш. (2022, 31 березня). *Що таке штучний інтелект?* – Повний посібник. HashDork. <https://hashdork.com/uk/what-is-artificial-intelligence>
- Дубінін, І. М. (1982). *Гармонія* (Ч. 2). Музична Україна.
- Кравчук, О. (2023). Застосування штучного інтелекту

в музичній індустрії України: аналітичний підхід. *Вісник Київського національного університету культури і мистецтв. Серія: Музичне мистецтво*, 6(1), 79–88. <https://doi.org/10.31866/2616-7581.6.1.2023.277888>

- Лемішко, М. М. (2010). *Гармонія* (2-ге вид.; Ч. 1: Діатоніка). Нова книга.
- Морева, Є. (2018). Сучасна гармонія у теоретичному і практичному курсах системи музичної освіти. *Наукові записки Тернопільського національного педагогічного університету імені Володимира Гнатюка. Серія: Мистецтвознавство*, 1(38), 5–10.
- Осадча, О. (2020). Роль штучного інтелекту в музичній творчості: психоаналітичний та метаантропологічний аналіз. В Н. Хамітов & С. Крилова (Ред.), *Людина і штучний інтелект: виміри філософської антропології, психоаналізу, арт-терапії та філософської публіцистики. Підхід філософської антропології як метаантропології* (с. 221–224). КНТ.
- Ракунова, І. М. (2008). *Нові композиторські технології (на прикладі творчості Алли Загайкевич)* [Автореферат дисертації кандидата мистецтвознавства, Національна музична академія України імені П. І. Чайковського].
- Римський-Корсаков, М. А. (1948). *Практичний підручник гармонії* (Є. Дроб'язко, пер.). Мистецтво.
- Фадеева, К. В. (2007). Музична комп'ютерна композиція: специфіка використання фрактальних алгоритмів. *Мистецтвознавчі записки*, 12, 22–31.
- Чібалашвілі, А. (2021). Штучний інтелект у мистецьких практиках. *Сучасне мистецтво*, 17, 41–50. <https://doi.org/10.31500/2309-8813.17.2021.248425>
- Dannenberg, R. B. (2000). Artificial intelligence, machine learning, and music understanding. In *Brazilian Symposium on Computer Music (SBCM2000)* [Proceedings of the Conference] (pp. 1–10). Brazil.
- Schoenberg, A. (1910). *Theory of Harmony* (R. E. Carter, Trans.). University of California Press.
- Targosz, J. (2004). *Podstawy harmonii funkcijnej*. Polskie Wydawnictwo Muzyczne.
- Zulić, H. (2019). How AI can change/improve/influence music composition, performance and education: Three case studies. *INSAM Journal of Contemporary Music, Art and Technology*, 1(2), 100–114.

References

- Babych, I. R., & Yashyna, O. M. (2020). *Pidkhody do rozrobky shtuchnoho intelek-tu ta yikhonii vplyv na avtomatyzatsiiu napysannia muzyky* [Approaches to the development of artificial intelligence and their impact on the automation of music composition]. *Modern Scientific Researches*, 12(1), 35–39 [in Ukrainian].
- Bkhatti, Sh. (2022, March 31). *Shcho take shtuchnyi intelekt?* – *Povnyi posibnyk* [What is artificial intelligence? –

- Complete guide]. HashDork. <https://hashdork.com/uk/what-is-artificial-intelligence> [in Ukrainian].
- Chibalashvili, A. (2021). Shtuchnyi intelekt u mystetskykh praktykakh [Artificial intelligence in artistic practices]. *Contemporary Art*, 17, 41–50. <https://doi.org/10.31500/2309-8813.17.2021.248425> [in Ukrainian].
- Dannenberg, R. B. (2000). Artificial intelligence, machine learning, and music understanding. In *Brazilian Symposium on Computer Music (SBCM2000)* [Proceedings of the Conference] (pp. 1–10). Brazil [in English].
- Dubinin, I. M. (1982). *Harmonii* [Harmony] (Pt. 2). Muzychna Ukraina [in Ukrainian].
- Fadieieva, K. V. (2007). Muzychna kompiuterna kompozytsiia: Spetsyfika vykorys-tannia fraktalnykh alhorytmiv [Musical computer composition: Specifics of using fractal algorithms]. *Notes on Art Criticism*, 12, 22–31 [in Ukrainian].
- Kravchuk, O. (2023). Zastosuvannia shtuchnoho intelektu v muzychnii industrii Ukrainy: Analitychnyi pidkhid [Application of artificial intelligence in the music industry of Ukraine: An analytical approach]. *Bulletin of Kyiv National University of Culture and Arts. Series in Musical Art*, 6(1), 79–88. <https://doi.org/10.31866/2616-7581.6.1.2023.277888> [in Ukrainian].
- Lemishko, M. M. (2010). *Harmonii* [Harmony] (2nd ed.; Pt. 1: Diatonika [Diatonics]). Nova knyha [in Ukrainian].
- Morieva, Ye. (2018). Suchasna harmoniia u teoretychnomu i praktychnomu kursakh systemy muzychnoi osvity [Contemporary harmony in the theoretical and practical courses of the system of music education]. *The Scientific Issues of Ternopil National Pedagogical Volodymyr Hnatiuk University. Series: Art Studies*, 1(38), 5–10 [in Ukrainian].
- Osadcha, O. (2020). Rol shtuchnoho intelektu v muzychnii tvorchosti: psykhoanalytychnyi ta metaantropolohichniy analiz [The role of artificial intelligence in musical creativity: psychoanalytic and meta-anthropological analysis]. In N. Khamitov & S. Krylova (Eds.), *Liudyna i shtuchnyi intelekt: Vymiry filosofskoi antropologii, psykhoanalizu, art-terapii ta filosofskoi publitsystyky. Pidkhid filosofskoi antropologii yak metaantropolohii* [Human and artificial intelligence: Dimensions of philosophical anthropology, psychoanalysis, art therapy, and philosophical journalism. The approach of philosophical anthropology as meta-anthropology] (pp. 221–224). KNT [in Ukrainian].
- Rakunova, I. M. (2008). *Novi kompozytorski tekhnologii (na prykladi tvorchosti Ally Zahaikevych)* [New composer technologies (on the example of Alla Zahaikevych's work)] [Abstract of PhD Dissertation, Ukrainian National Tchaikovsky Academy of music] [in Ukrainian].
- Rymskyi-Korsakov, M. A. (1948). *Praktychnyi pidruchnyk harmonii* [Practical textbook of harmony] (Ie. Drobiazko, Trans.). Mystetstvo [in Ukrainian].
- Schoenberg, A. (1910). *Theory of Harmony* (R. E. Carter, Trans.). University of California Press [in English].
- Targosz, J. (2004). *Podstawy harmonii funkcijnej* [Fundamentals of functional harmony]. Polskie Wydawnictwo Muzyczne [in Polish].
- Zulić, H. (2019). How AI can change/improve/influence music composition, performance and education: Three case studies. *INSAM Journal of Contemporary Music, Art and Technology*, 1(2), 100–114 [in English].

Prospects of Software Application for Generating Extended Chords and Their Modifications

Andrii Bondarenko

Kyiv National University of Culture and Arts, Kyiv, Ukraine

The aim of the article is to analyse the capabilities of software for generating and studying extended chords and their modifications. *Results.* It is revealed that the study of extended chords has not received sufficient coverage in the scientific and educational literature. The article demonstrates that the software environment enables the development of software capable of generating extended chords and their modifications (inversions and different voicings). The number of possible extended chords depends on the number of notes and ranges from 4 (for triads) to 37 (for 10-note chords), and then decreases to 4 for 12-note chords. It is shown that the 9th chord can accept 120 modifications, the 11th chord — 720, while the 13th chord — 5040. However, both for the convenience of performers and for stylistic reasons, when developing software, it is advisable to provide the option of limiting the range and interval structure of a given chord. *The scientific novelty* lies in the in-depth understanding of the inversion and arrangement of extended chords, as well as the pioneering use of a C++ software environment for the development of chord generation software capable of handling complex structures. For the first time, a complete table of 9th chord modifications and a table of all possible 12-note

extended chords are presented. *Conclusions.* It is shown that the development of appropriate software can enhance the understanding of artists and musicologists regarding the possibilities of extended chords, facilitate their work, and open up new horizons of creativity. The obtained results are useful for updating the educational and methodological literature on harmony, which currently does not sufficiently reveal the theory of extended chords. However, it is concluded that the software cannot replace the act of creativity itself, which requires the artist's emotional involvement, as well as multi-level contextual thinking when selecting harmonic elements.

Keywords: harmony; extended chords; composition; artificial intelligence

Інформація про автора

Андрій Бондаренко, кандидат мистецтвознавства, старший викладач, Київський національний університет культури і мистецтв, Київ, Україна, ORCID iD: 0000-0002-6856-991X, e-mail: bondareandre@gmail.com

Information about the author

Andrii Bondarenko, PhD in Art Studies, Senior Lecturer, Kyiv National University of Culture and Arts, Kyiv, Ukraine, ORCID iD: 0000-0002-6856-991X, e-mail: bondareandre@gmail.com

